



المدرسة الوطنية العليا في الأمن السيبراني
NATIONAL SCHOOL OF CYBERSECURITY

1ST YEAR BASIC TRAINING IN CYBER SECURITY

INTRODUCTION TO OPERATING SYSTEMS 1 (SYST1)

Dr. Sassi BENTRAD

✉ : sassi.bentrad.enscs@gmail.com || sassi.bentrad@enscs.edu.dz

LISCO Laboratory (Laboratoire d'Ingénierie des Systèmes COMplexes) / University of Badji Mokhtar-Annaba (UBMA)
National School of Cyber Security (NSCS)

Basic Training in Cyber Security (1BT)
Formation de Base en Cyber-Sécurité (1FB)



CHAPTER

2

UNIX/LINUX OPERATING SYSTEM BASICS

SYST1'2025/2026



COURSE CONTENT

CHAPTER 2

UNIX/LINUX OPERATING SYSTEM BASICS (15 %)

- ❑ Presentation of the UNIX/Linux system
- ❑ UNIX/Linux System structure
- ❑ UNIX/Linux Main functions
- ❑ Command language :
 - ✓ *Linux Shell*
 - ✓ *Most famous and popular shells*
 - ✓ *Shell command prompt*
 - ✓ *Shell usage modes & Modification*
- ❑ Package Managers
- ❑ System Connection & Sessions
- ❑ UNIX / Linux Users

❖ PRESENTATION OF UNIX/LINUX

□ Origins

- Unix or **UNIX** was originally called **UNICS** (**Un**iplexed **I**nformation **C**omputing **S**ystem) is a robust, *multi-user, multitasking operating system.*
- In the **1960s** and **1970s** Dennis Ritchie and Ken Thompson invented **UNIX**, arguably **the world's most important computer operating system.**
- **UNIX** was born at **BELL Laboratories** (a subsidiary of **AT&T : American Telephone and Telegraph Company**).
- UNIX was first developed in **Assembly language**.
- From **1973**, **UNIX** is **90%** recoded (rewritten) in **C language**



A woman enters copy using Unix.



❖ PRESENTATION OF UNIX/LINUX

□ Origins

- In 1975, the **UNIX** source-code was distributed to universities.
- Development of **2 branches**:
 - ✓ **BSD** developed at the University of Berkeley (California)
 - ✓ **System V** sold by AT&T to Sun Microsystems, IBM, DEC and HP
- **UNIX®** is a registered trademark since 1994



Ken Thompson (L) and Dennis Ritchie (R)



❖ PRESENTATION OF UNIX/LINUX

□ UNIX Timeline



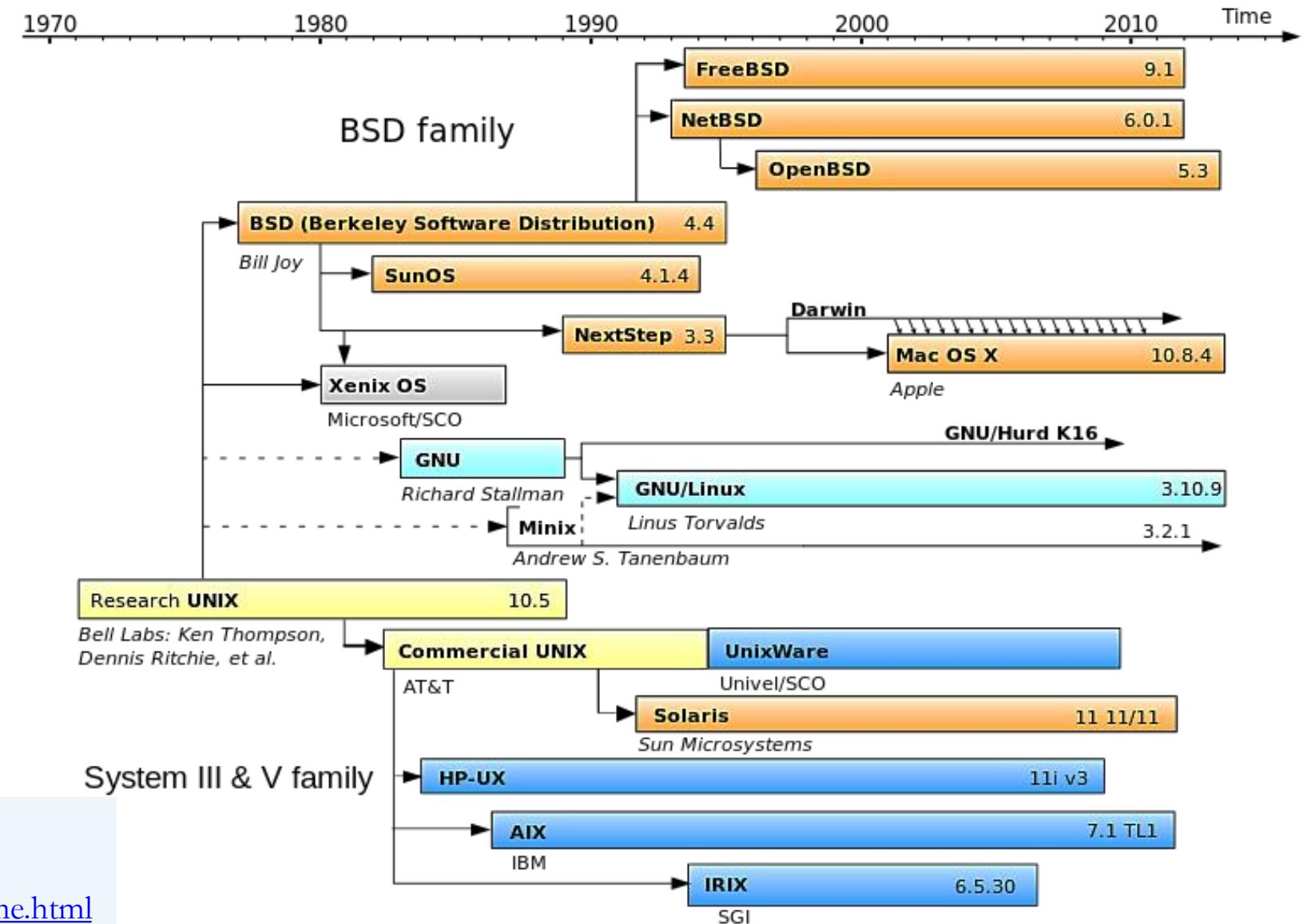
**Evolution of Unix
and Unix-like
systems**

1969 -- 2010

History and Timeline :

<https://www.levenez.com/unix/>

https://unix.org/what_is_unix/history_timeline.html



2. UNIX/LINUX OPERATING SYSTEM BASICS

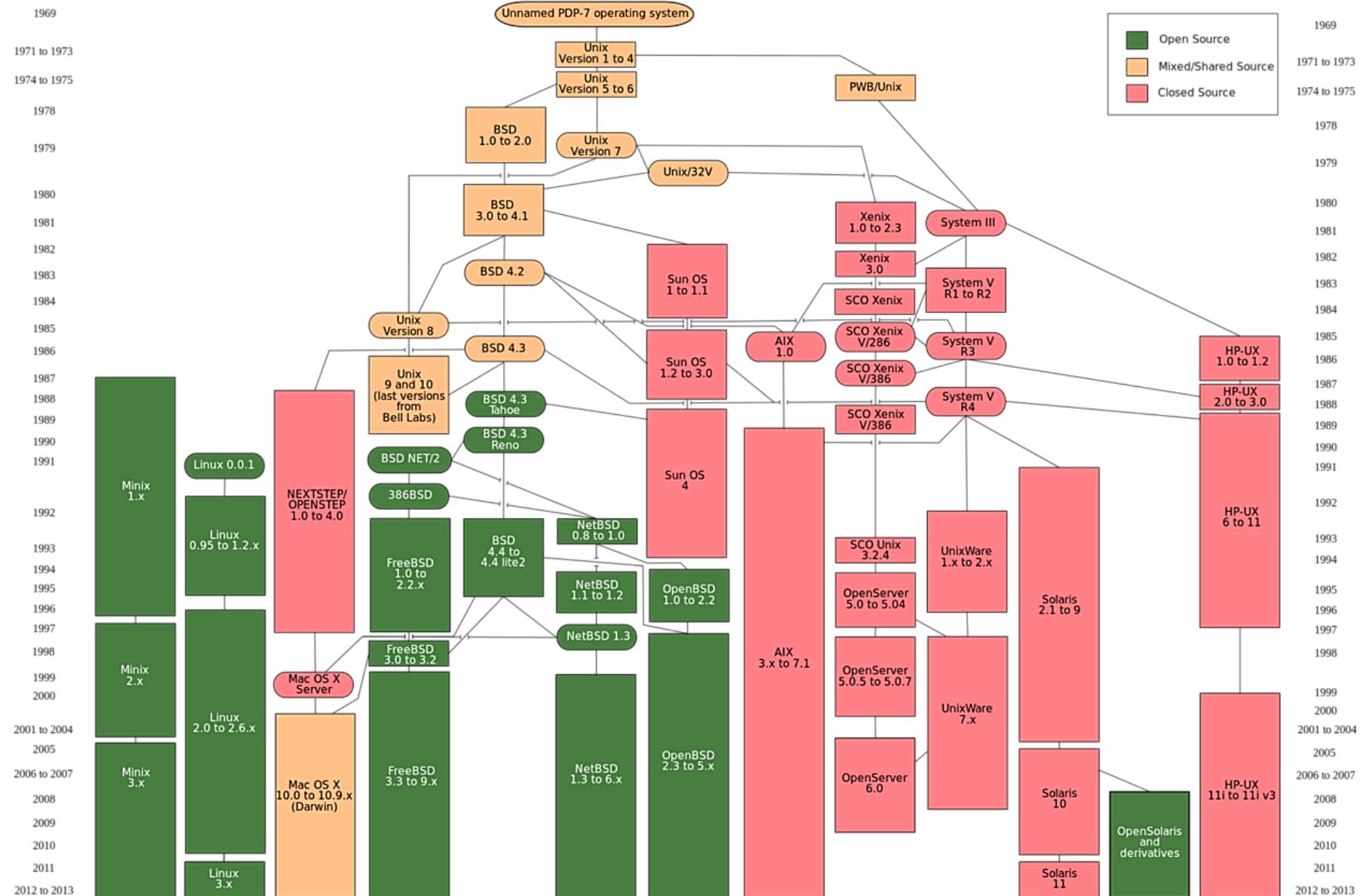
❖ PRESENTATION OF UNIX/LINUX

□ UNIX Timeline



THE **Open** GROUP

**Evolution of Unix
and Unix-like
systems**
1969 -- 2013

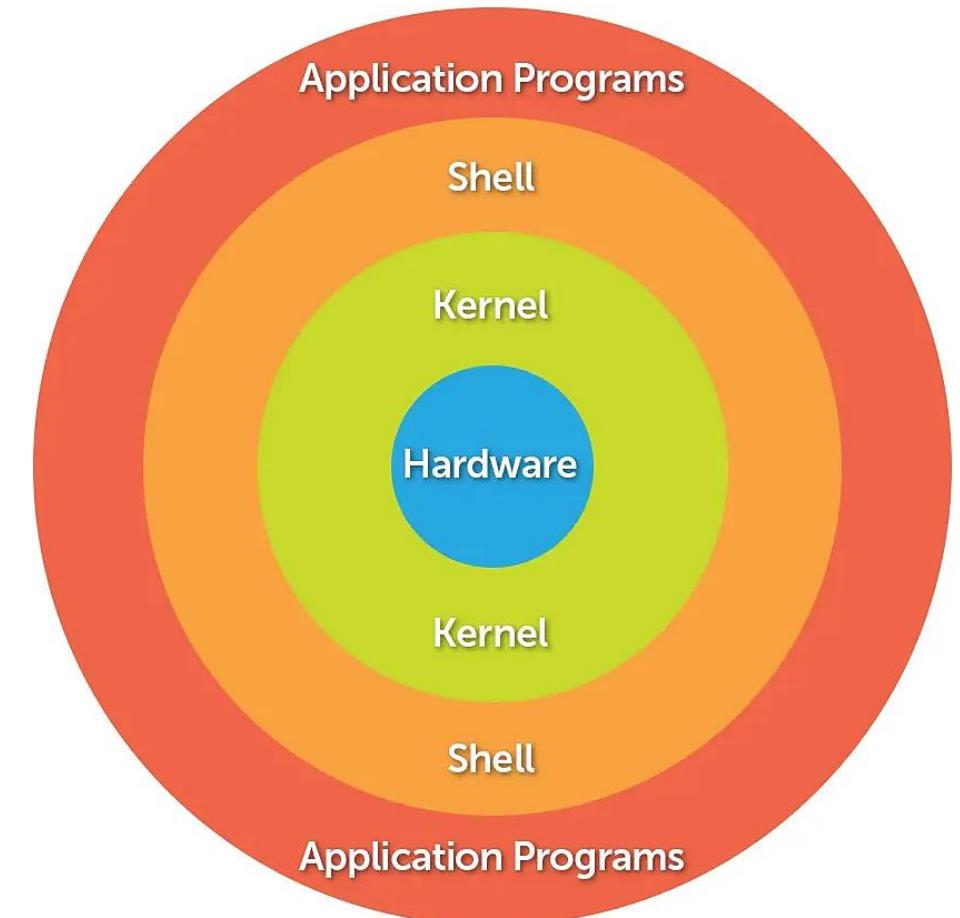


❖ PRESENTATION OF UNIX/LINUX

□ System Architecture

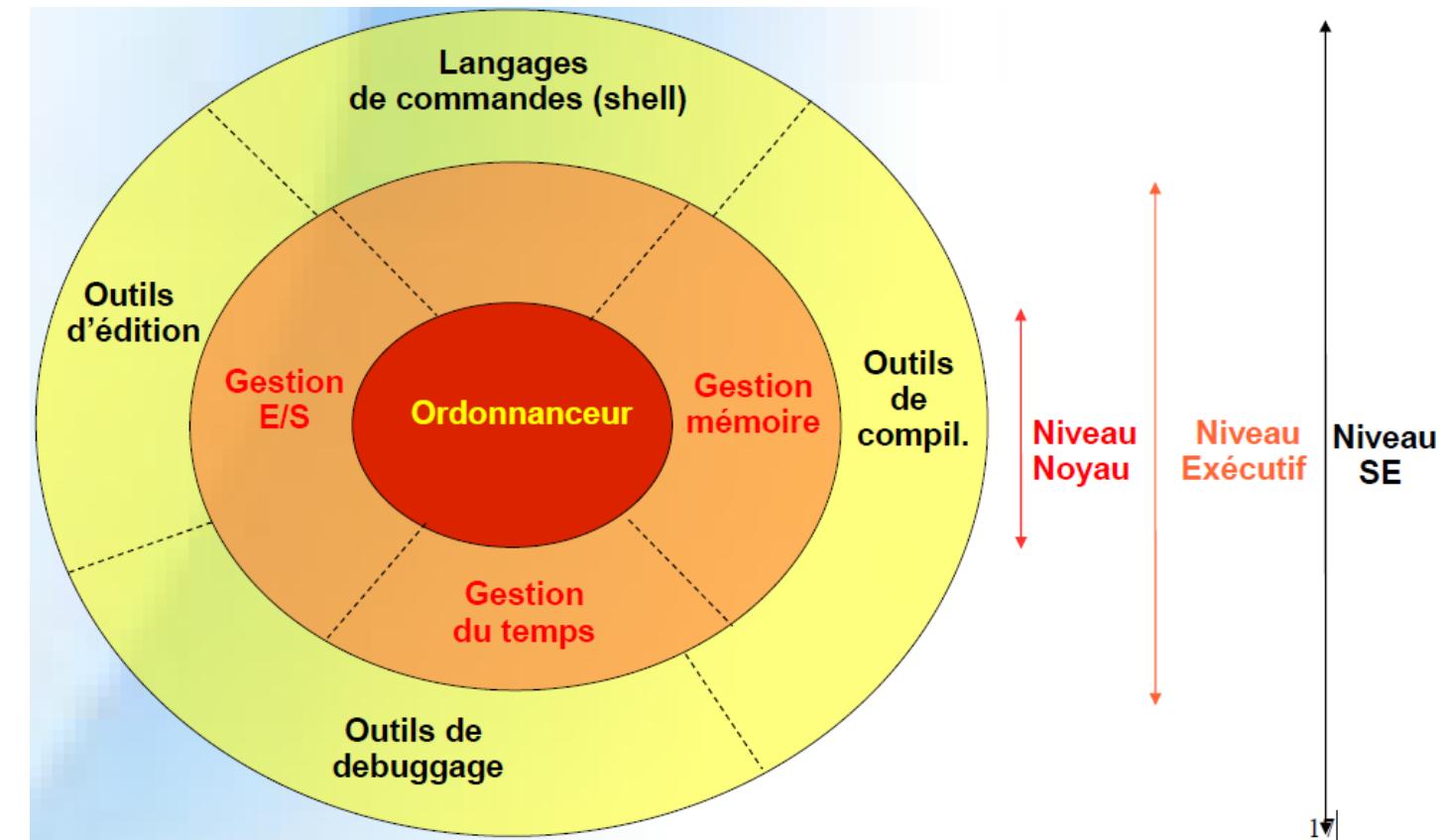
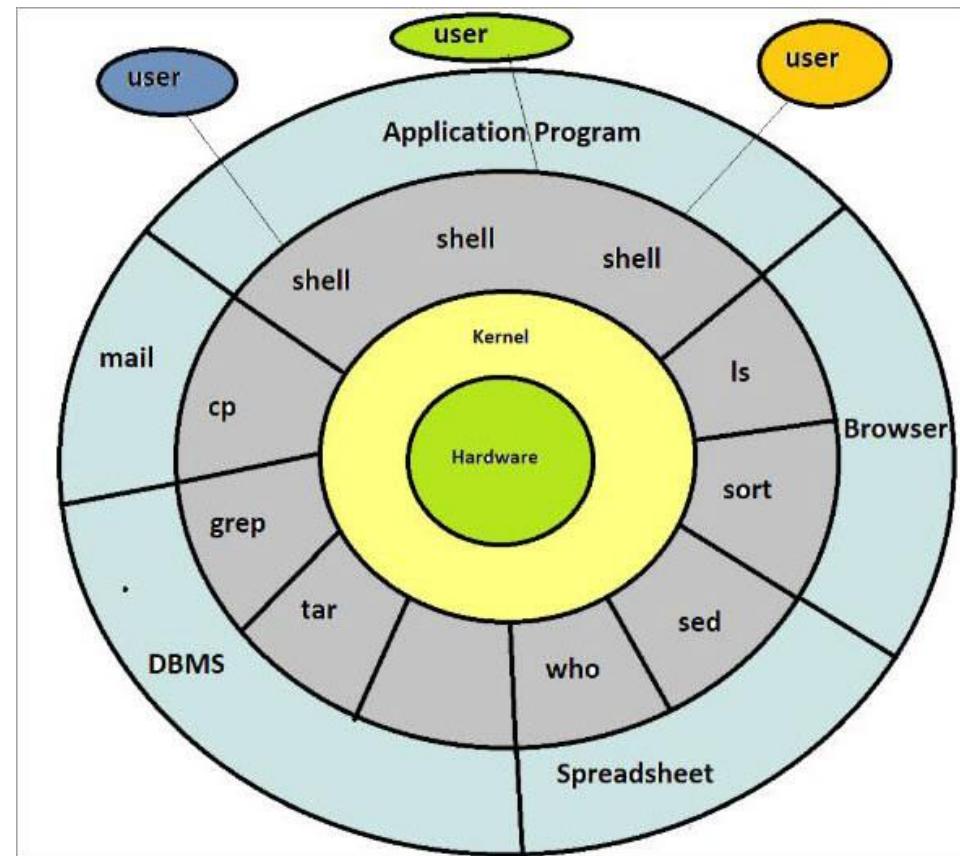
The **architecture of the Unix operating system** is divided into **four layers**.

All four layers work in tandem with each other to handle complex tasks efficiently.



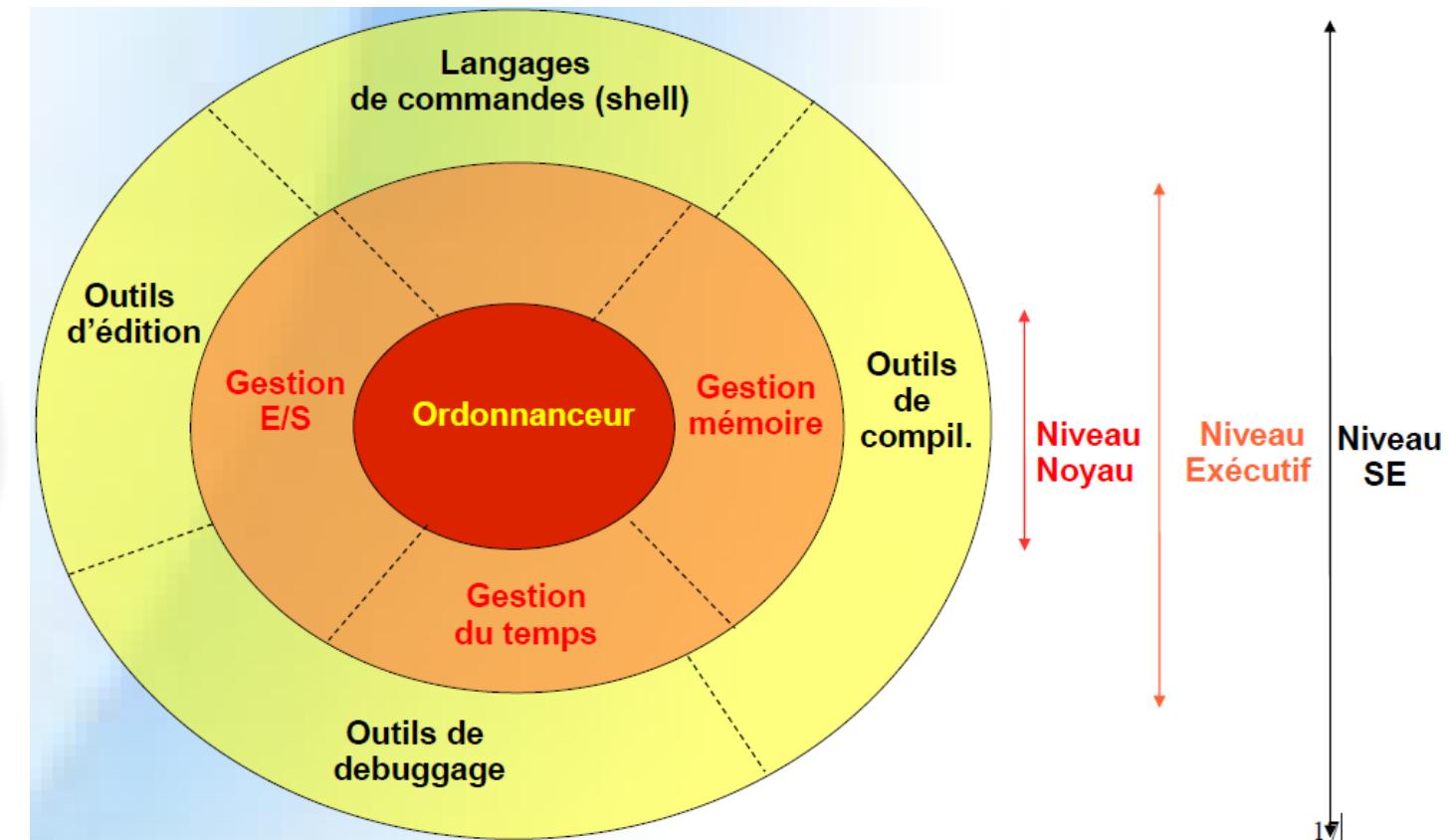
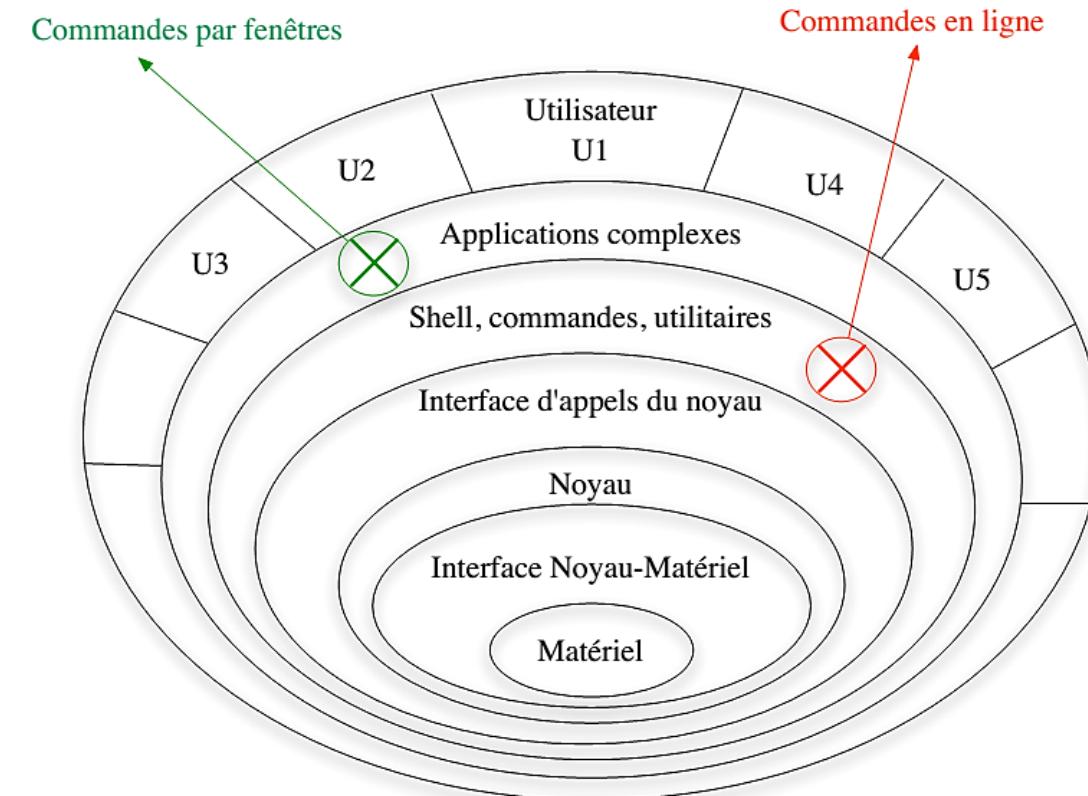
❖ PRESENTATION OF UNIX/LINUX

□ System Architecture



❖ PRESENTATION OF UNIX/LINUX

□ System Architecture



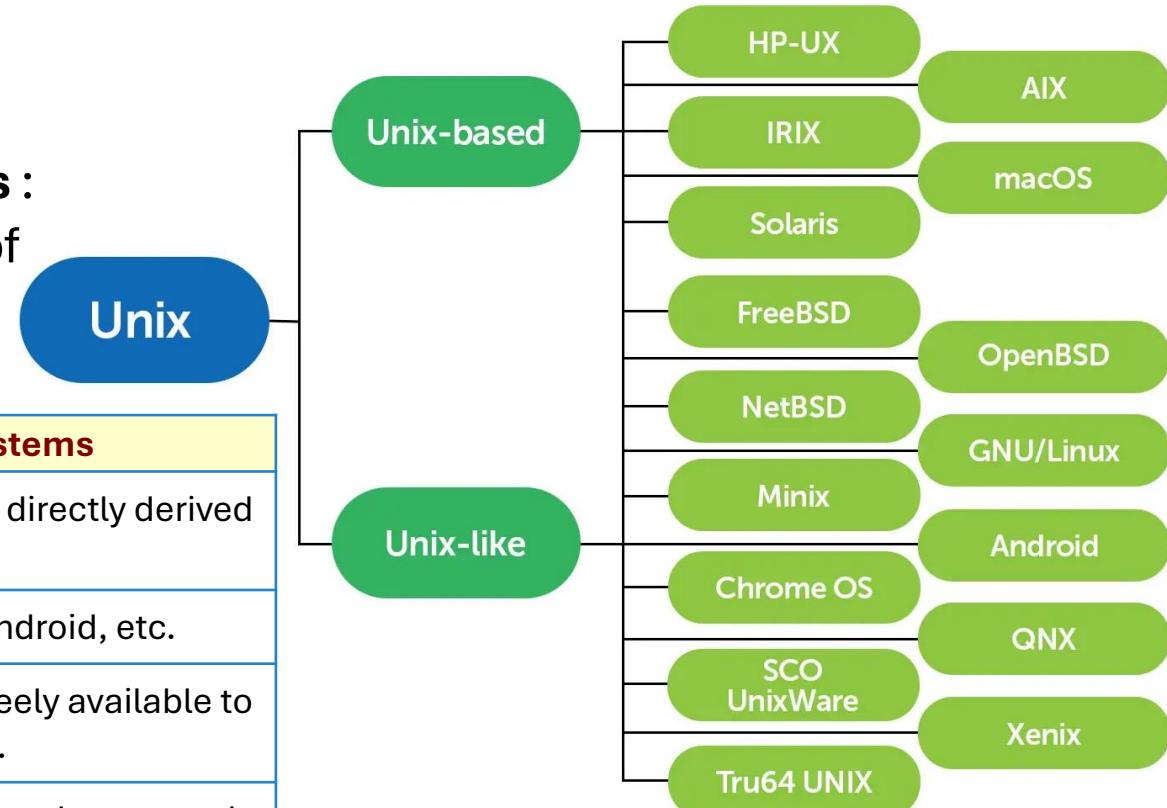
❖ PRESENTATION OF UNIX/LINUX

□ Types of Unix Operating Systems

Unix operating systems can be **categorized** into **two types** :

Unix-based systems and **Unix-like** systems. The names of these systems are quite self-explanatory.

Feature	Unix-based Systems	Unix-like Systems
Definition	Directly derived from the original Unix system.	Similar to Unix, but not directly derived from it.
Examples	macOS, Solaris, AIX, etc.	Linux, FreeBSD, Android, etc.
Source Code	Often proprietary, not always open to the public.	Mostly open-source, freely available to anyone.
User Base	Used by large companies and enterprises (servers, workstations).	Used by individuals, developers, and companies (especially for personal computers and servers).
Customization	Limited customiz., controlled by the vendor (e.g., Apple).	Highly customizable, many different versions (distributions) are available.
User Interface	Graphical user interfaces (e.g., macOS's GUI).	Can have both command-line (CLI) or graphical interfaces (particularly Linux).



❖ PRESENTATION OF UNIX/LINUX

□ Use Cases of Unix Operating System

The **Unix operating system** has widespread applications across various sectors and industries.

Here are the common uses :

- **Mission-critical applications**
- **Scientific computing**
- **Web Servers**
- **Database Servers**
- **Embedded Systems**
- **Desktops**
- **Telecommunications**



❖ PRESENTATION OF UNIX/LINUX

□ LINUX Distributions



GNU Operating System

<https://www.gnu.org/>

- **GNU** (Gnu's Not Unix) is a **huge collection of programs and utilities** that was started by **Richard Stallman**.
- In **1992**, the installation of a **Linux system** remained an adventure reserved for an elite of tinkerers with a certain sense of adventure.
- A **distribution** is the packaging of Linux. The **kernel** (developed by **Linus Torvalds**) **remains the same on all distributions**.
- A **Linux distribution** = **Kernel + Installation tools + Administration tools + Applications package**.

Why several distributions ?

- Focused on ease of use;
- For real geeks ;
- For use in schools, etc.
- For old (low-performance) PCs.

FREE SOFTWARE FOUNDATION

<https://www.fsf.org/>



❖ PRESENTATION OF UNIX/LINUX



GNU/Linux Alternatives

- Liberté d'exécuter le logiciel, pour n'importe quel usage ;
- Liberté d'étudier et de modifier le fonctionnement d'un programme ;
- Liberté de redistribuer des copies ; et
- Obligation de faire bénéficier à la communauté des versions modifiées.

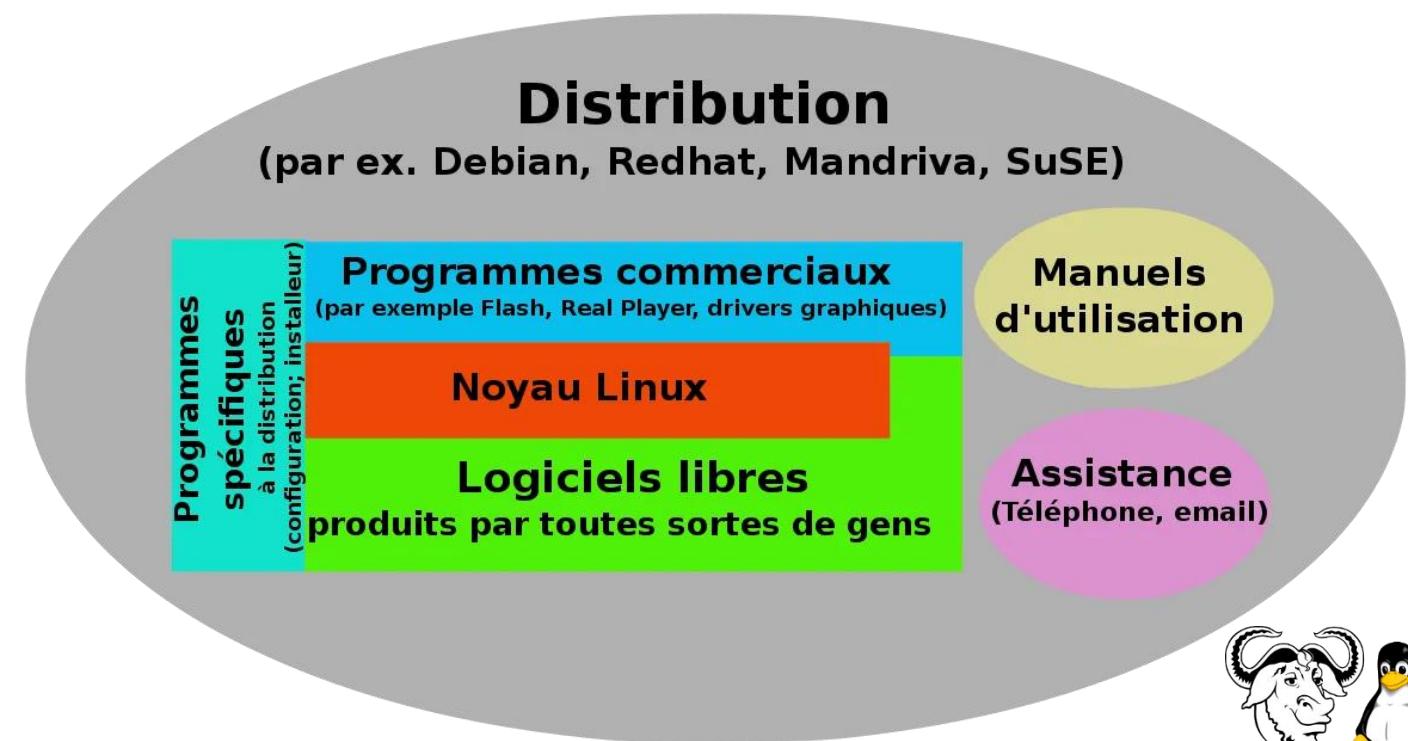


Linux - The Penguin  marches On



<https://www.linuxfoundation.org/>

The **LF** is a non-profit organization, provides a **neutral, trusted hub** for developers to code, manage, and scale open technology projects.

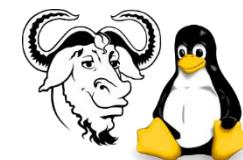
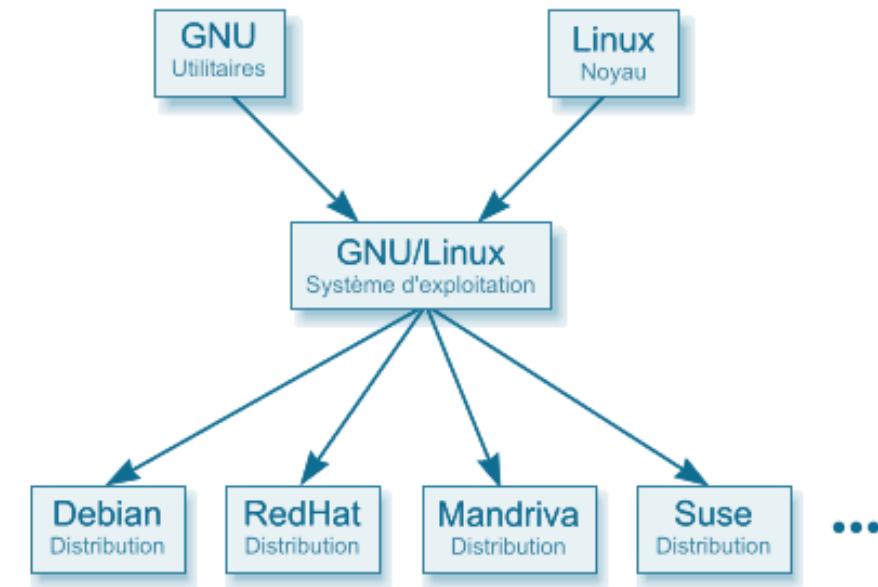


❖ PRESENTATION OF UNIX/LINUX

❑ LINUX Distributions

- In 1993, Slackware Linux (*popular with Geeks*).
- In 1994, Red Hat (*primarily for servers*).
- In 1996, Debian GNU/Linux (*Linux Experts*).
- In 1998, Mandriva (*Domestic use*).
- In 2002, Ubuntu, based on the Debian distribution.
 - ✓ Package manager, manages all installed software and makes it easy to uninstall.
 - ✓ A new version released every six months.
 - ✓ Easy program search and installation.
 - ✓ ...

A GROWING WORLD OF LINUX DISTRIBUTIONS



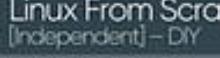
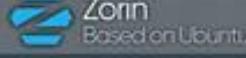
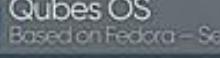
❖ PRESENTATION OF UNIX/LINUX

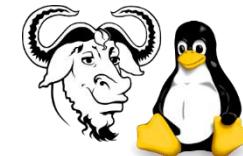
□ Overview of different types of LINUX distributions



LINUX distributions can be broadly categorized into **three main types**:
Desktop distributions, **Server distributions**, and **Embedded distributions**.

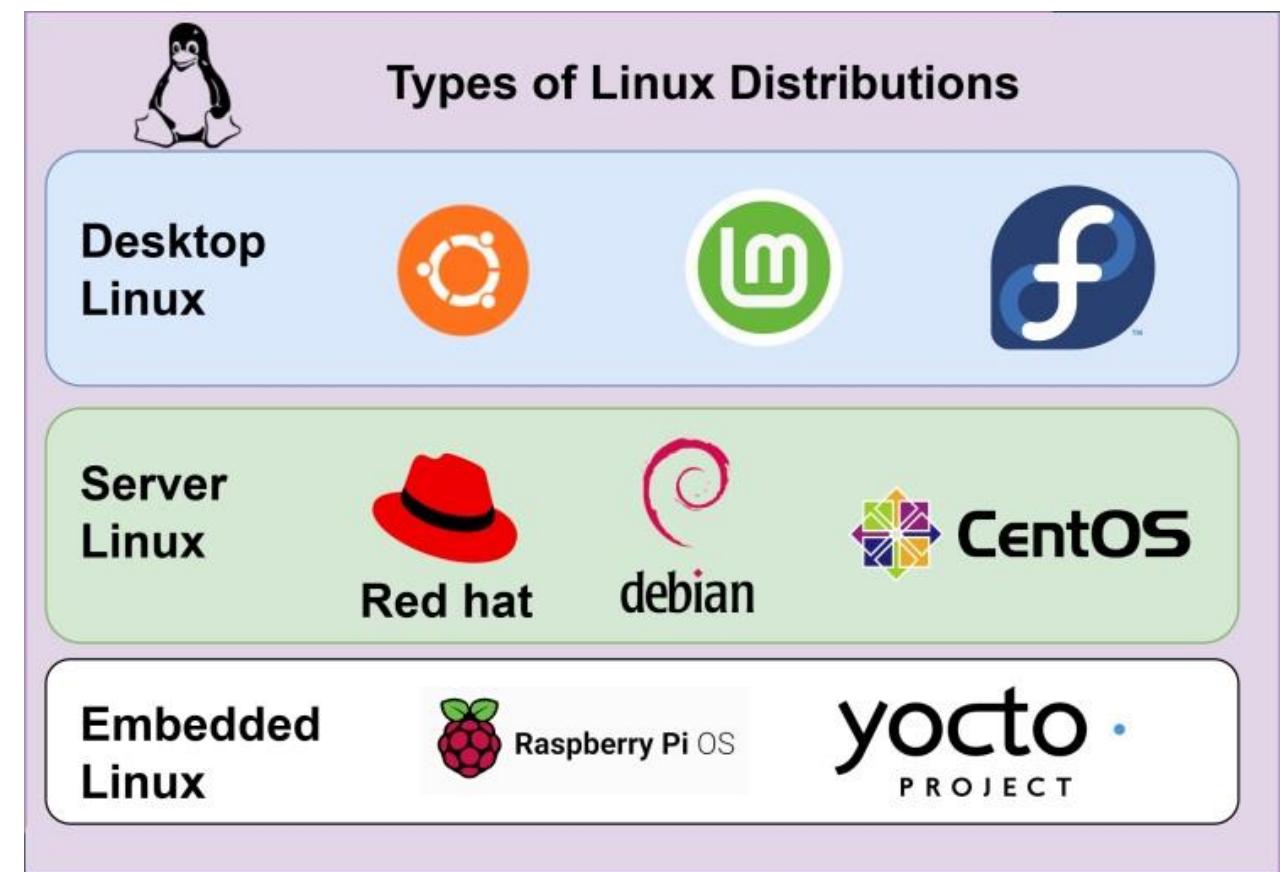
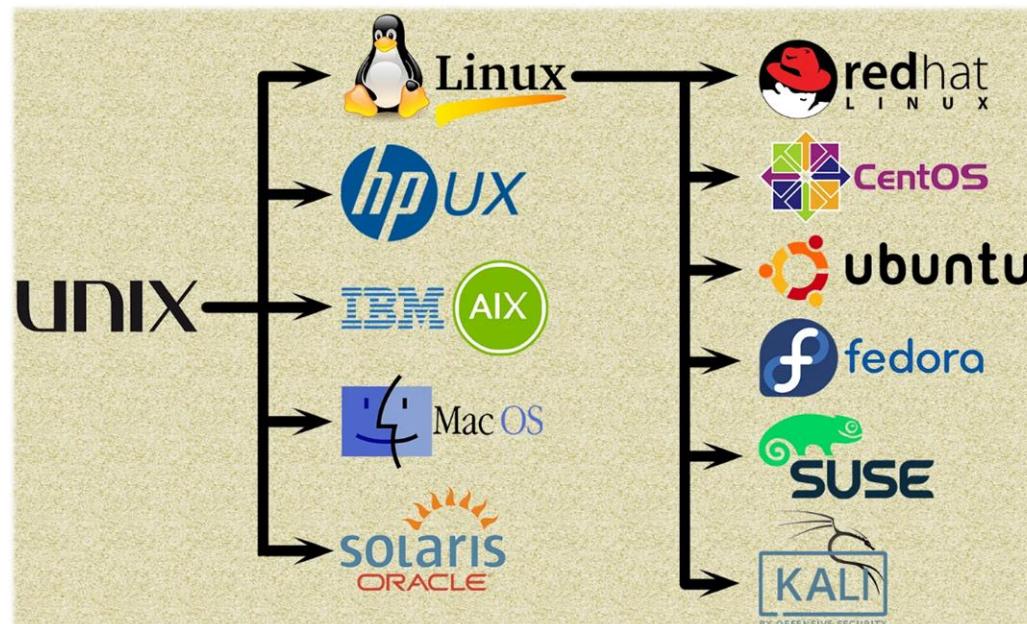
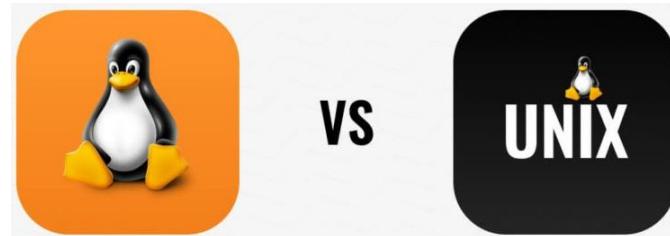
Linux Distros : www.distrowatch.com

Beginner-friendly	Intermediate	Hard mode
 Ubuntu Based on Debian	 Garuda Linux Based on Arch	 Arch [Independent] – DIY
 Pop!_OS Based on Ubuntu	 EndeavourOS Based on Arch	 Gentoo [Independent] – DIY
 elementary OS Based on Ubuntu (LTS)	 Manjaro Based on Arch	 Slackware [Independent]
 Mint Based on Ubuntu	 MX Linux Based on Debian	 Linux From Scratch [Independent] – DIY
 Zorin Based on Ubuntu	 Fedora Based on Red Hat	 Qubes OS Based on Fedora – Security
 Solus [Independent]	 OpenSUSE [Independent]	 NixOS [Independent] – DIY



❖ PRESENTATION OF UNIX/LINUX

❑ UNIX vs. Linux



❖ PRESENTATION OF UNIX/LINUX

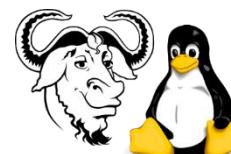
□ Key Features of LINUX



- **LINUX** offers the **same functions and features** as **UNIX**
- **Hardware Quality** : Versatile, can run on low-spec PCs
- **Cost** : Mostly free, some distributions have paid versions
- **Software Compatibility** : Open-source substitutes for proprietary software
- **Ease of Installation** : Requires some computing knowledge
- **Security and Stability** : The safest and most stable OS
- **Ease of Use** : Ease of use determined by the distribution.

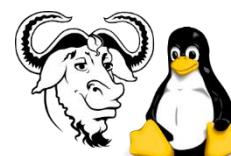
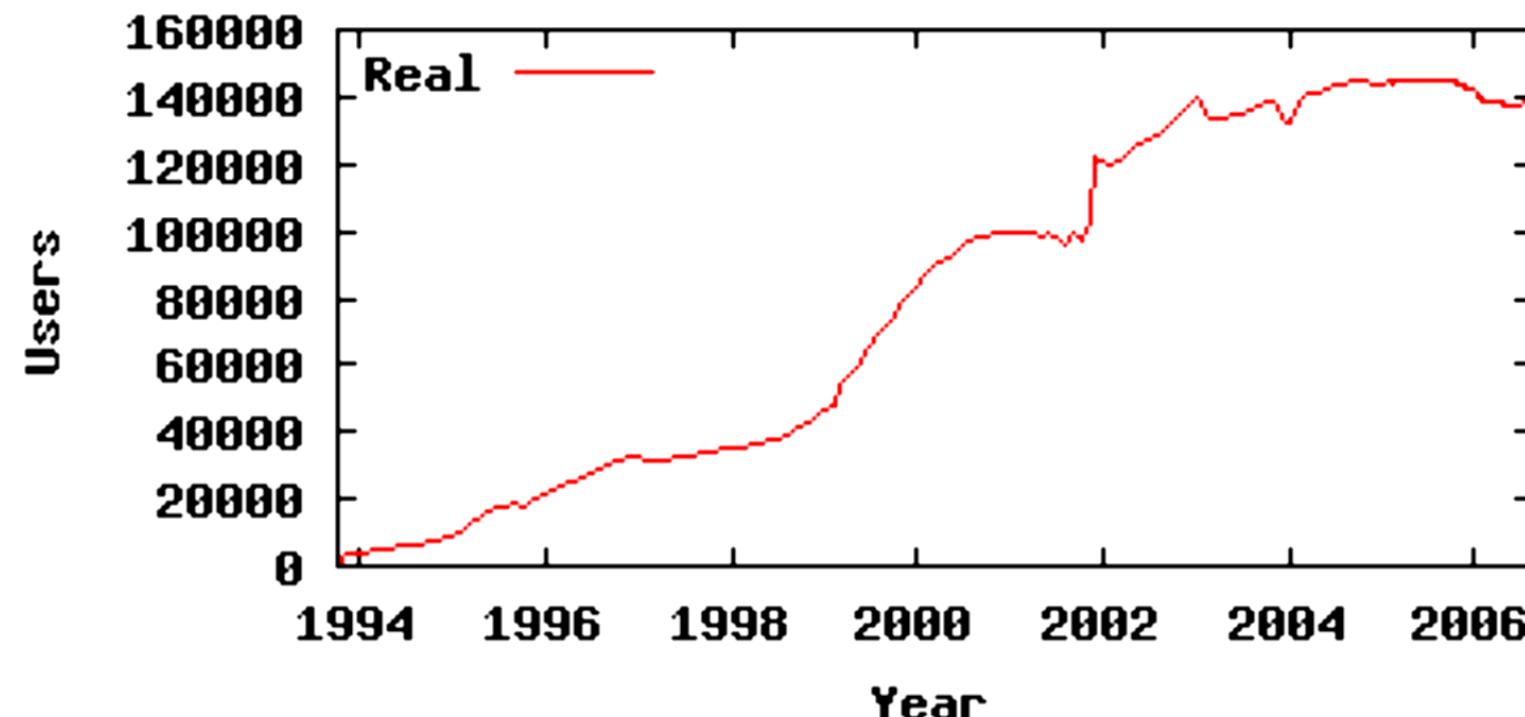
Linux is the grandson of **UNIX**.....

BUT it is increasingly converging to take the place of **UNIX**.



❖ PRESENTATION OF UNIX/LINUX

Linus made **LINUX** a **free code** at any change by the end of **1992**, there were **100 LINUX developers**, this number is multiplied every year.



❖ PRESENTATION OF UNIX/LINUX

□ Which LINUX Distro to Choose ?

Choosing a **Linux distribution** can be one of **the most difficult things for a Linux user**. There are so many excellent options, and they all have their own unique strong suits.

There are also constant updates, news, and general community chatter that muddy the waters even more, making the process much less of a direct route.

However, there are **a couple of questions that you can ask yourself** in choosing a distribution that help clear things up. **It's also important to remember that there usually isn't a wrong answer.**

Every distribution is good. Sure, running Arch on your enterprise scale production deployment probably isn't the best idea, but it's still technically possible. It's all about picking a distribution that fits around that sweet spot of what you want and need.



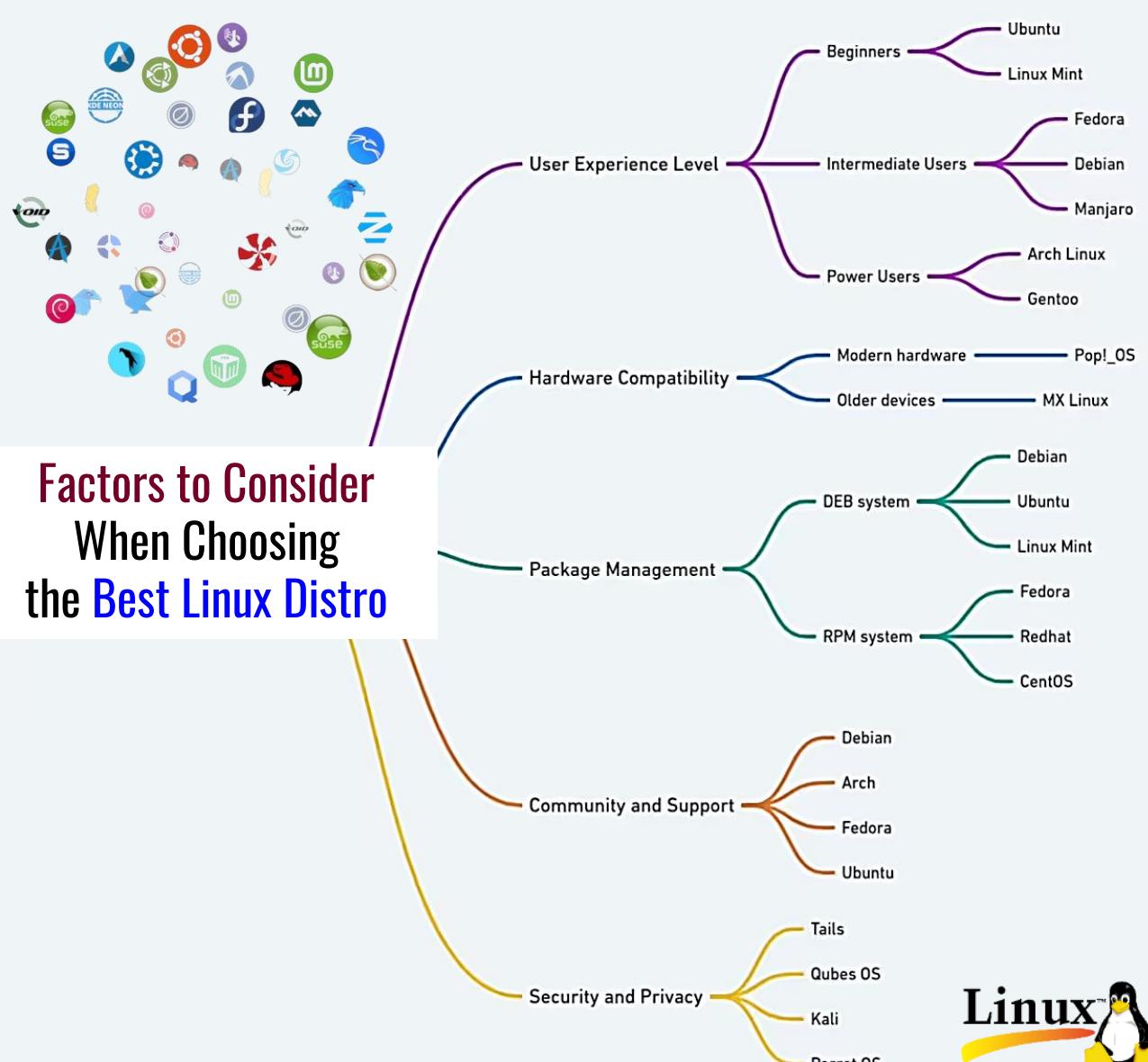
❖ PRESENTATION OF UNIX/LINUX

□ Which LINUX Distro to Choose ?

How to Choose Guide for Every User

The **best Linux version** for you is the one that meets your personal requirements.

Linux's **flexibility** allows you to explore, experiment, and find the distro that fits your workflow and lifestyle best.



❖ PRESENTATION OF UNIX/LINUX

❑ Which LINUX Distro to Choose ?

- ✓ **The best Linux version for you** is the one that meets your personal requirements.
- ✓ **Linux's flexibility** allows you to explore, experiment, and find the distro that fits your workflow and lifestyle best.

**Factors to consider when choosing the Best Linux Distribution
for personal or commercial use.**

<https://linuxconfig.org/best-linux-distro-how-to-choose-guide-for-every-user>



❖ PRESENTATION OF UNIX/LINUX

□ UNIX/Linux Features

- **Multi-platforms** : independent of any hardware architecture.
- **Time-sharing multitasking** : Run multiple programs at the same time.
 - ✓ Multi-tasking is ensured by a pre-emptive mechanism: the system authoritatively interrupts the task in progress to hand over to the next.
- **Multi-users** : Multiple users can use the machine at the same time.
 - ✓ The simultaneous cohabitation of several users is made possible by an access rights mechanism that applies to all the resources managed by the system (processes, files, devices, etc.).
- **Open operating system** : Its operating principles are known in detail and it is well documented.



UNIX®

The word 'UNIX' is written in a large, bold, green, sans-serif font. A registered trademark symbol (®) is located at the top right corner of the letter 'X'. There is a thin horizontal line above the 'U' and another below the 'X'.

❖ PRESENTATION OF UNIX/LINUX

□ UNIX/Linux System Connection

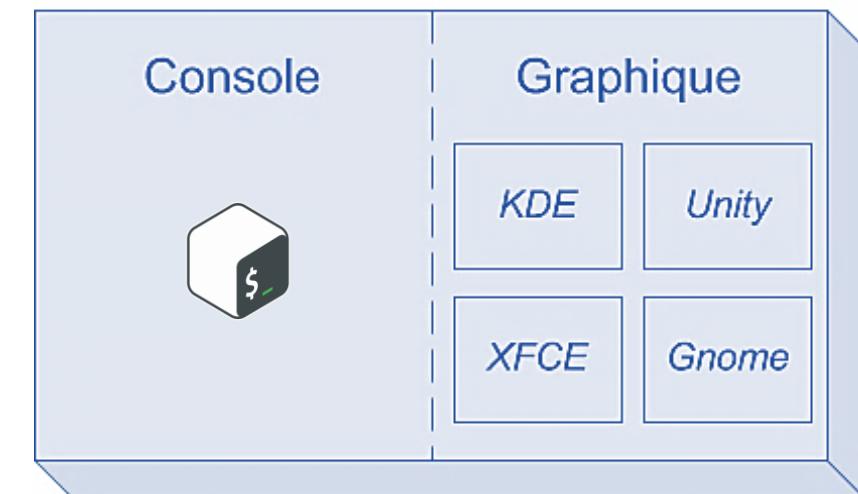
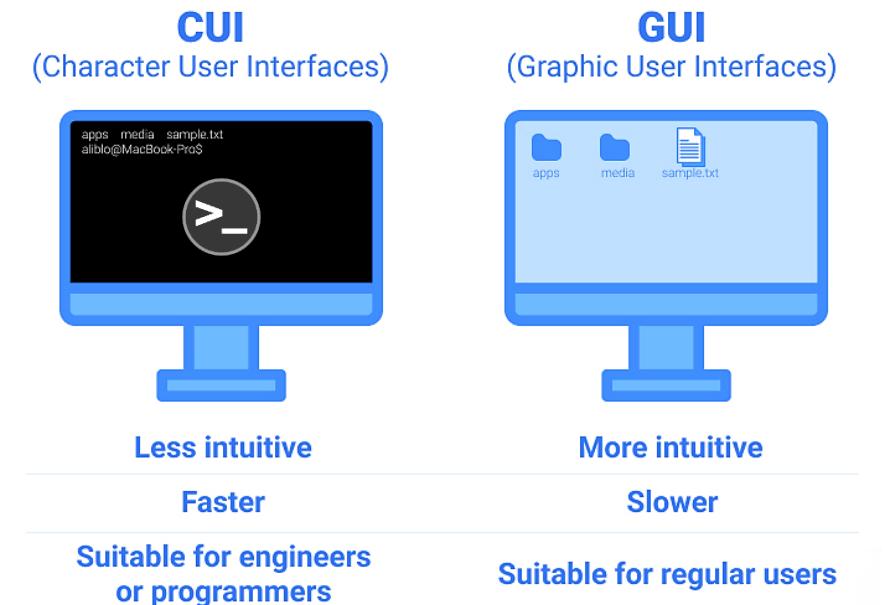
- You have to log in to be able to work on **Linux**
- There are **two connection modes** :

✓ Connection in **text-based or console mode** (**terminal**): execution only of commands without graphical mode.

CLI (Command-Line Interface)

✓ Connection in **graphical mode (X-Window)**: use of the mouse, graphical windows, launch applications...

GUI (Graphical-User Interface)



❖ BASIC FUNCTIONS

■ Process Management

- Execution Sequence (**Process**)
- Hardware Event Management

■ Information Management

- Memory Management
- File & Directory Management

■ Communication management

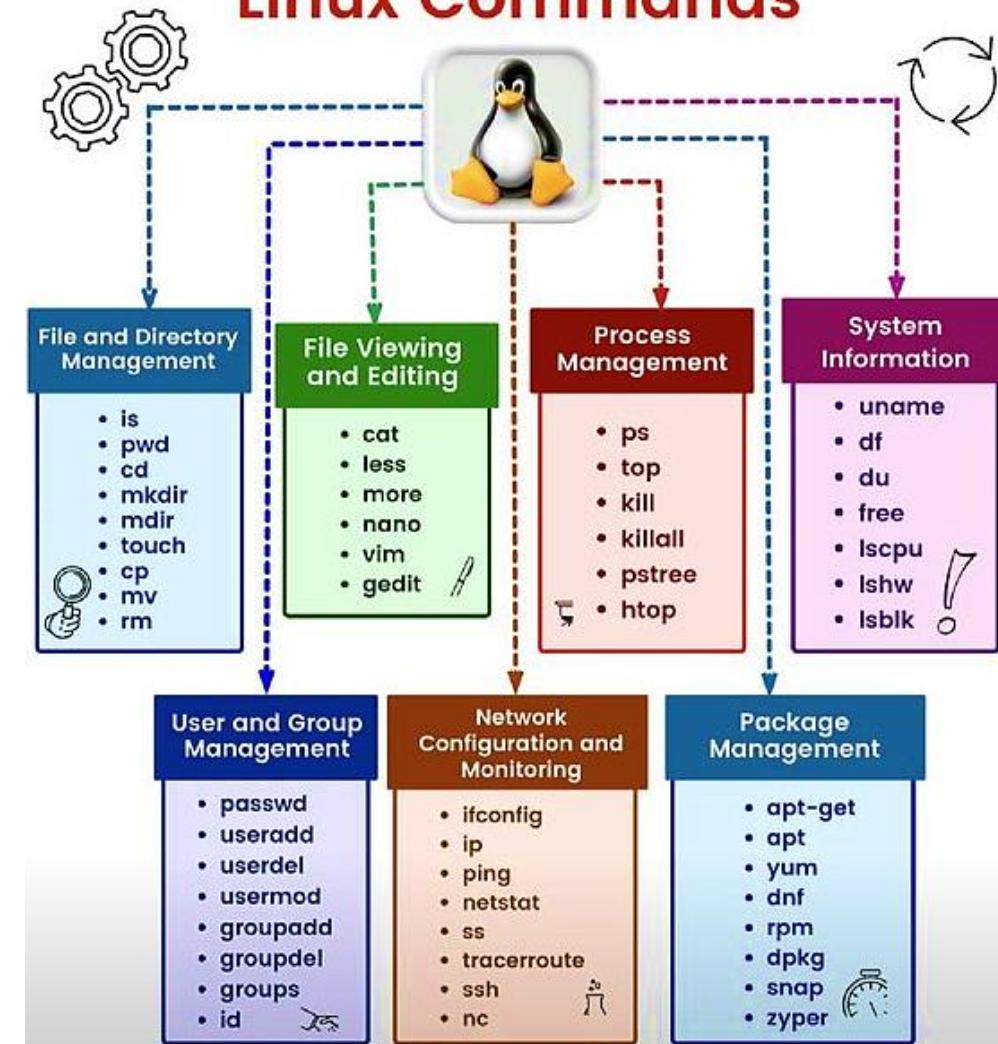
- Input/Output Management
- Network Configuration and Monitoring

■ Security and Protection

- Information Security and Protection
- Resource Security and Protection

■ ...

List of Important Linux Commands



❖ COMMAND LANGUAGE

□ The Shell

The **Shell** serves as a **command interpreter** between users and the kernel. When you enter **commands** into your system, it's the **shell** that interprets these instructions for execution by the **kernel**.

Once tasks are completed, it facilitates displaying results back to the user.

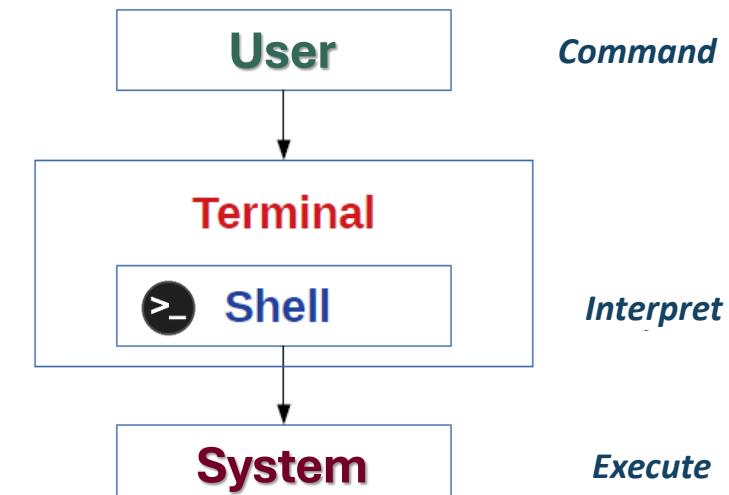
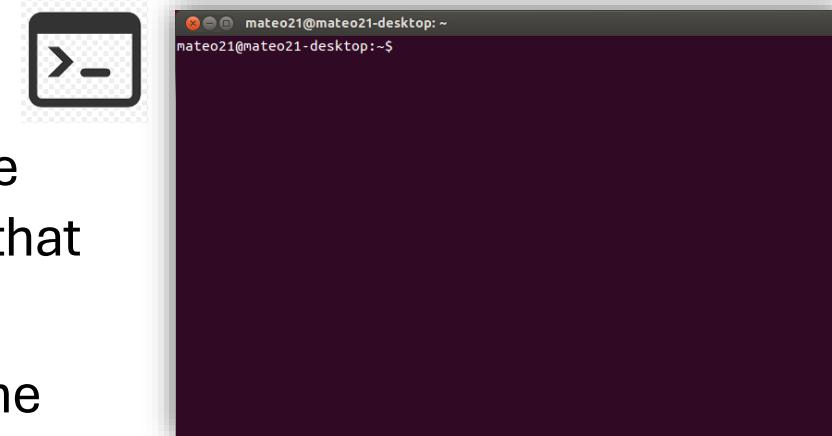
A **command** is an **instruction** given to the system through the **terminal** in order to execute a specific action.

The **terminal** is a program used to execute commands.

Each line of the **terminal** is of the form:

User@machine:directory\$

Example : user@PC:~\$



Command Execution Process

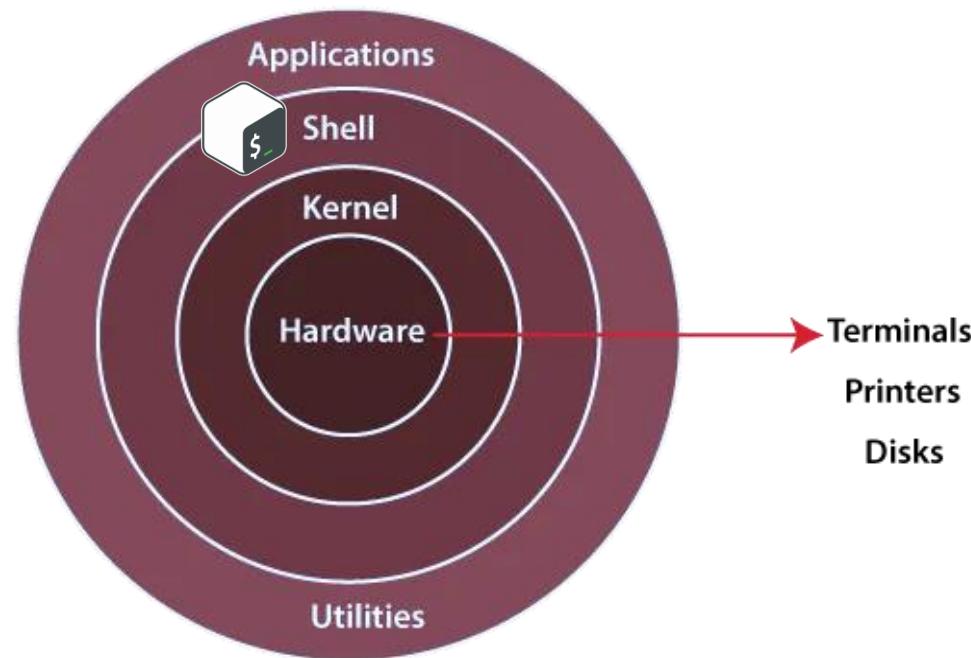
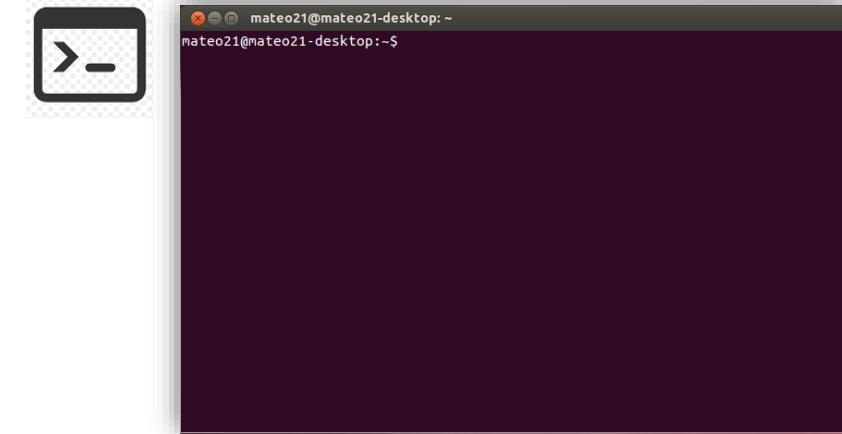
❖ COMMAND LANGUAGE

□ The Shell

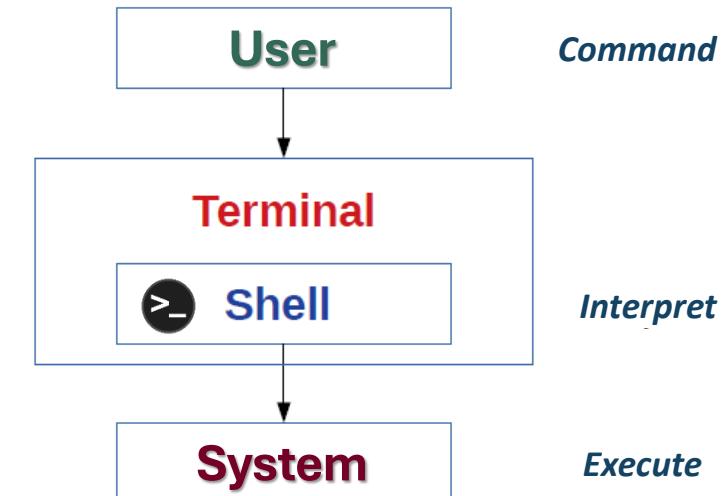
Some **Linux distributions** install several shells.

These might include :

gnome-terminal, konsole, xterm, rxvt, kvt, nxterm, and eterm.



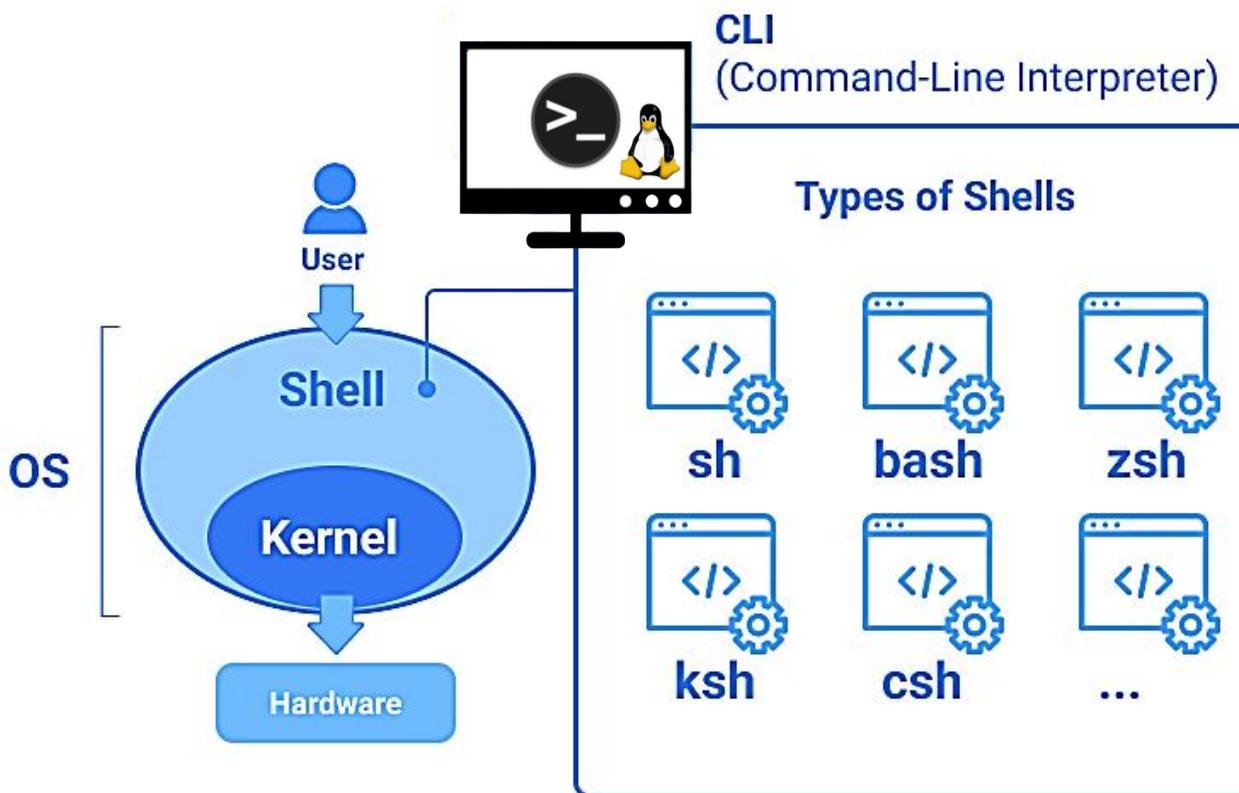
Architecture of Linux System



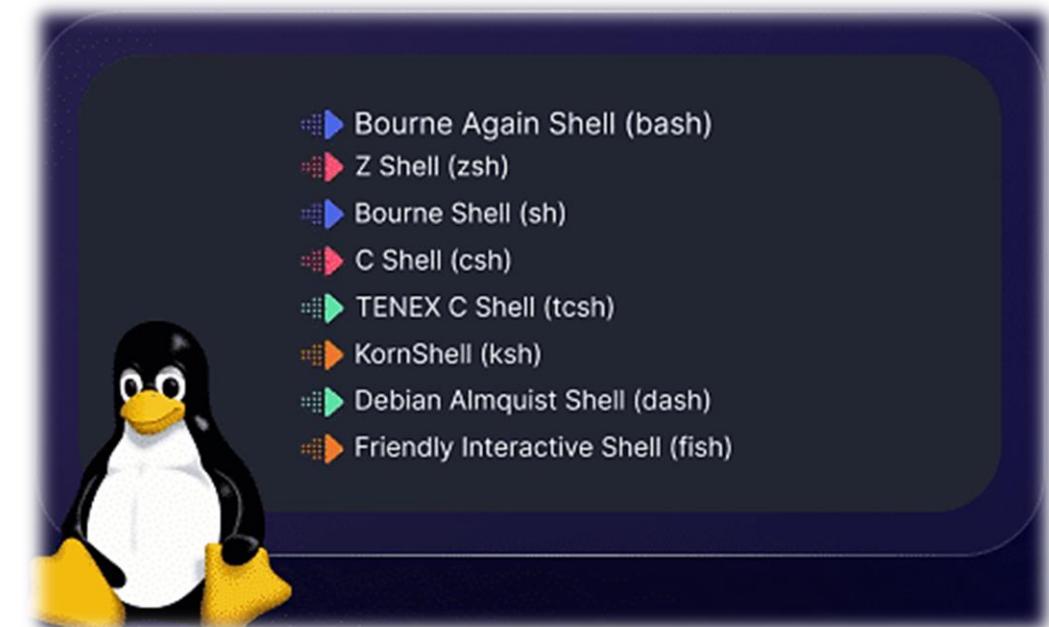
Command Execution Process

❖ COMMAND LANGUAGE

❑ Kernel & Shell



Most famous and popular shells



Note: Checking your current Shell

`echo $SHELL`



❖ COMMAND LANGUAGE

❑ Importance of Linux Shell



- **Efficiency** : The shell lets you complete critical tasks quickly with text commands that get the job done faster than a graphical interface.
- **Scripting and Automation** : Shells are a critical component in system automation that uses scripts to automate repetitive tasks.
- **Remote Access** : The shell is the primary way to access and control remote Linux systems.
- **Economic Resource Consumption** : Shells use fewer system resources compared to graphical interfaces.



❖ COMMAND LANGUAGE

❑ Most famous and popular Shells



- **Bourne Shell** and its derivatives :
 - ✓ **sh** : bourne **shell** (obsolete), Traditional, basic shell found on Unix systems, by Steve Bourne.
 - ✓ **ksh** : korn **shell** is derived from the *sh*.
 - ✓ **zsh** : **Z shell**, quite recent and contains the best ideas from *bash*, *ksh* and *tcsh*.
- **C Shell** and its derivatives :
 - ✓ **csh** : The **C shell** (obsolete), Once popular shell with a *C-like syntax*. developed by Berkeley University.
 - ✓ **tcsh** : The **TC shell** (still very popular), compatible implementation with evolved features (command completion, history editing and more...)
- **Bash**: The **Bourne Again shell** (most popular), an improved implementation of *sh* with lots of added features too.

Many commands are common to different Shell's



❖ COMMAND LANGUAGE

❑ Most famous and popular Shells

Friendly Interactive Shell (**fish**) : (*a great new shell*)

<https://fishshell.com/>



```
root@Server:~# fish
Welcome to fish, the friendly interactive shell
Type help for instructions on how to use fish
root@Server ~#
root@Server ~# cd /home
root@Server /home#
root@Server /home#
```



fish is a **smart** and **user-friendly command line shell**.

fish is built for an **easy** and **interactive command-line experience**.

It features intuitive syntax, robust auto-completion, and helpful error messages for beginners and experienced users.



❖ COMMAND LANGUAGE

❑ Most famous and popular Shells

Friendly Interactive Shell (fish) : (*a great new shell*)

<https://fishshell.com/>



Getting fish : Packages for Linux

Packages for **Debian**, **Fedora**, **openSUSE**, and **Red Hat Enterprise Linux/CentOS** are available from the [openSUSE Build Service](#).

Packages for **Ubuntu** are available from the [fish PPA](#), and can be installed using the following commands:

```
sudo apt-add-repository ppa:fish-shell/release-4
```

```
sudo apt update
```

```
sudo apt install fish
```



 COMMAND LANGUAGE Most famous and popular Shells Friendly Interactive Shell (**fish**)**Key Features :**

- 1. Auto-suggestions:** As you type commands, Fish will show suggestions based on history and commands previously executed.
- 2. Syntax Highlighting:** Fish automatically highlights syntax as you type.
- 3. No Need for Configuration Files:** Fish is designed to work out of the box without requiring a .bashrc or .zshrc file. Its configuration is stored in a `~/.config/fish/` directory.
- 4. Powerful Scripting:** Fish uses a more readable scripting syntax compared to bash.
- 5. Web-based Configuration:** Fish provides a built-in web-based interface for configuring the shell. By running `fish_config` in the terminal, you can access the config UI in your browser.



❖ COMMAND LANGUAGE

❑ Most famous and popular Shells Friendly Interactive Shell (**fish**)



Key Features :

5. **Clean and Consistent Syntax:** Fish's syntax is intuitive and consistent.
6. **Tab Completion:** Fish offers advanced, user-friendly tab completion that provides suggestions as you type, not just for commands but for paths and options as well.
7. **Universal Variables:** These variables persist across all sessions and are automatically shared between different Fish shells.
8. **Easy-to-Use Prompt:** Fish has a clean and attractive default prompt with a fish icon, but you can easily customize it.
9. **Cross-Platform:** Fish works on macOS, Linux, and Windows via **WSL** (Windows Subsystem for Linux).



❖ COMMAND LANGUAGE

□ Shell Command Prompt

Prompt - The **shell** displays a prompt at the beginning of the line, called a **prompt** ('\$' or '#' or '%'), to tell the user that they are waiting for a command to be entered.

Example : `admin@hp : ~/Cours/OS1$`

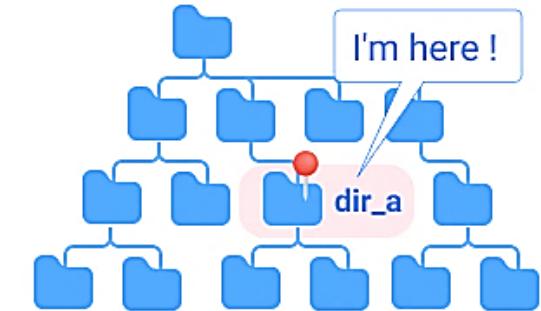
- Username: admin
- Machine Name: hp
- Current Working Directory: `~/Cours/OS1$`
- Prompt Symbol (user Indicator at the end) :

\$: Ordinary user

: Superuser

\$: *sh, bash, ksh*
% : *csh, zsh*

Directory Tree and Current Working Directory on GUI



Current Working Directory on CUI



❖ COMMAND LANGUAGE

□ Shell usage modes

✓ Simple & Powerful **Command Line Interpreter (CLI) (interactive mode)**



✓ Interpretive programming language (**scripts**)



✓ **Fish Shell – Friendly Interactive Shell**

🔄 Changing Default Shell

To set **Fish** as your default shell:

chsh -s /usr/bin/fish

Log-out and back-in for the change to take effect.



Tip

Fish is not always located in **/usr/bin/fish**.

Check its path using:

which fish

and use that path in the **chsh** command.



❖ PACKAGE MANAGERS

A **package manager** is a tool for automating the process of installing, upgrading, configuring, or removing programs in a consistent manner.



□ Role of Package Managers

- Systems contain many interdependent software packages and libraries that require regular updates.
- Manual management of these dependencies is complex and error-prone.
- **Package managers** automate installation, updates, and dependency handling, ensuring software compatibility and keeping the system up to date.



❖ PACKAGE MANAGERS

□ How Package Managers Update a Computer



- In a **CUI** environment, package managers are used through **specific commands**.
- These tools **connect to repositories**, **download updates**, and **install newer versions** of software automatically.
- Each **Linux distro** uses **its own package manager** with unique commands.

- Automate the process of **installation**, **upgrading**, **configuring**, or **removing** of programs
- Trigger the process with **commands** (`apt`, `yum`, `rpm`, etc.)



❖ PACKAGE MANAGERS

❑ Linux OS package managers



- Linux distros use two main types of package managers : **Debian-based** and **RPM-based**.
- Each type has its own **command syntax** and **tools**.
- Common examples of package managers for each distribution type are listed below.

Category	Distributions	Package Manager Tools (Commands)
Debian-based	Debian, Ubuntu	apt (apt-get) dpkg
RPM-based	RHEL, CentOS, Fedora, OpenSUSE	rpm yum dnf



Basic Training in Cyber Security (1BT)
Formation de Base en Cyber-Sécurité (1FB)



CHAPTER

2

A large, stylized number '2' is centered within a light blue circular frame, indicating the chapter number.

UNIX/LINUX OPERATING SYSTEM BASICS

- *SYSTEM CONNECTION & SESSIONS*
- *UNIX / LINUX USERS*

❖ SYSTEM CONNECTION & SESSIONS

□ System Connection

◆ Common Process

A **session** begins with **user identification** followed by **authentication**.

- Enter **username** at the login prompt.
- Enter **password** at the password prompt.

◆ Switching Between Modes

Linux allows switching between **Graphical Mode (GUI)** and **Console Mode (CLI)** using keyboard shortcuts :

- **Ctrl + Alt + F1 ... F6** → Switch to console terminals.
- **Ctrl + Alt + F7** → Return to the graphical interface.



This flexibility enables users to manage or troubleshoot the system even if the graphical interface fails.



❖ SYSTEM CONNECTION & SESSIONS

❑ Logging In and Logging Out of a Session

◆ Logging In

Accessing a Linux account involves **two main steps**:

Entering your **Username**

Entering your **Password**

Example:

<hostname> login: os1user

Password:

When typing the password, **no characters appear** on the screen — this is a **security measure** to prevent others from seeing it.

If the username or password is incorrect, the system displays : **Login incorrect**

and restarts the login process.

◆ Logging Out

To **end a session**, type the command :

logout or **exit**

This safely ends your session and returns to the login prompt.

or close the terminal window
(or by pressing **Ctrl-D**).



*Always **log out** after completing your work to maintain system security.*



❖ SYSTEM CONNECTION & SESSIONS

□ Creating a User Account

◆ Step 1: Create the User

At the **shell prompt**, type the following command : **adduser nscsStudent**

Example output :

```
[root@localhost /root]# adduser nscsStudent  
[root@localhost /root]#
```

◆ Step 2: Assign a Password

Set a password for the new user by typing : **passwd nscsStudent**



Notes :

- When typing the password, **no characters appear** for security reasons.
- You must **retype the password** to confirm it.
- The message “*all authentication tokens updated successfully*” confirms that the user account has been created successfully.



❖ SYSTEM CONNECTION & SESSIONS

□ A Session / User Account

◆ Logging In to a Session

To log in, you must have a **user account** on the system.

◆ A User Account Includes :

- A **Username (Login)** linked to a unique **User ID (UID)**
- A **Confidential Password** for authentication
- A **Group** defining permissions and access
- A **Home Directory** to store personal files
- A **Shell** to interact with the system

◆ System Information :

- Account details are stored in the file: **/etc/passwd**
- Passwords are **encrypted** for security.
- The **administrator** can **reset** a password but **cannot view** the original one.

 *Each user account ensures security, personalization, and controlled access within the Linux OS.*



❖ SYSTEM CONNECTION & SESSIONS

□ Logging a session

Text-Based Connection (**CLI**) :

If the **login / password** are valid then :

user@machine :~#

→ *Logged in as root (administrator)*

user@machine :~\$

→ *Logged in as a regular user*

```
Ubuntu 18.04 ubuntu tty1
ubuntu login: Ubuntu
Password:
Welcome to Ubuntu 18.04 (GNU/Linux 4.15.0-23-generic)

 * Documentation: https://help.ubuntu.com/
278 packages can be updated.
71 updates are security updates.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

Ubuntu@ubuntu:~$
```

□ Closing a session

◆ **Purpose** : Properly closing a session is important for security and to **free system resources**.

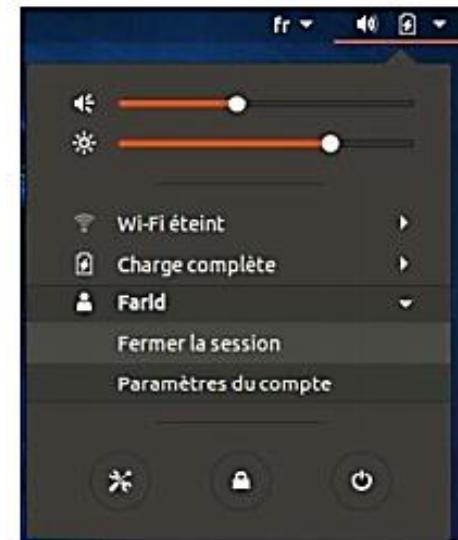
◆ Steps to Close a Session

- **Quit all applications** currently running.
- **Log out** of your session when finished using the system.
- This ensures that your **account and workstation** remain secure.

◆ Methods to Log Out

From the Graphical Interface (GUI):

Select “**Log Out**” from the system menu.



From the Console (CLI):

Use the key combination **Ctrl + Alt + F1, F2, ..., or F7** to switch terminals.

Type the command: **exit** or **logout**

❖ SYSTEM CONNECTION & SESSIONS

❑ Creating / Changing a password

◆ Why It Matters ?

A **strong password** is essential for **protecting personal data** and maintaining **system security**.

◆ Characteristics of a Strong Password

- Difficult to guess or predict
- At least **8 characters** long (preferably more)
- Includes a mix of:
 - ✓ **Uppercase letters (A-Z)** **Lowercase letters (a-z)**
 - ✓ **Numbers (0-9)** **Special characters (!, @, #, \$, %, etc.)**
- Does **not contain personal information**
- Is **not a dictionary word** or common phrase
- Avoids **repeated or sequential characters**
- Is **unique** for each account
- Is **easy to remember** but **hard for others to guess**

◆ Best Practices

- Change your password regularly.
- Immediately update it if you suspect unauthorized access.

Change the password with the **passwd** command.

Syntax : passwd [identifiant]

Identifiant : username

Only the **root user (administrator)** can change or reset the passwords of **other users**.

```
:~$ passwd farid
Changement du mot de passe pour farid.
Mot de passe UNIX (actuel) :
```



*A **secure password** is your first line of defense against system intrusion.*



*Regular users can update their own passwords, while only **administrators** can manage others' credentials.*

❖ SYSTEM CONNECTION & SESSIONS

□ Becoming Root (Superuser)

◆ Purpose

Some tasks, like **system administration or configuration**, require **superuser (root) privileges**.

By default, users operate with **limited permissions** for security reasons.

◆ Using the su - Command

To switch to the root account, type: **su -**

Enter the **root password** when prompted.
You'll gain full administrative privileges.

When finished, type: **exit**
to return to your normal user session.



Use root privileges with caution — actions performed as root can affect the entire system.

◆ Using the sudo Command

Instead of switching users, you can run a single command with root privileges:

sudo <command>

In **Ubuntu** and similar distributions, sudo is preferred because the root account is **disabled by default**.

Only users listed in the **/etc/sudoers** file can use **sudo**.

To learn more :

man sudoers



❖ SYSTEM CONNECTION & SESSIONS

□ Shutting Down from the Command Line

◆ Overview

Linux systems can be safely shut down directly from the **command line interface (CLI)**. Only the **root user** or users with **sudo privileges** can perform shutdown operations.

◆ Method 1: Using the **halt** Command : Logs out all users and powers off the system.

halt

◆ Method 2: Using the **shutdown** Command : The shutdown command provides more control over how and when to stop the system.

Examples :

Shutdown and power off immediately: **shutdown -h now**

Shutdown and reboot immediately: **shutdown -r now**

Schedule a shutdown: **shutdown -h +10** *# Shuts down after 10 minutes*



Always use **proper shutdown commands** to avoid data loss and ensure system integrity.



❖ UNIX / LINUX USERS

◆ Multi-User System :

- UNIX/Linux are designed as **multi-user operating systems**, allowing **multiple users** to work on the same machine simultaneously.

◆ User Ownership :

- Every **resource** (file, directory, or program) belongs to a **specific user** registered on the system.
- Access rights and permissions depend on **user ownership** and **group membership**.

◆ User Identification :

- Each user has a unique **Username (login)** and **Password** for authentication.
- The system assigns every user a **User Identifier (UID)** — a unique number that distinguishes one user from another.

```
admin@hp: ~$ id  
uid=3069(admin) gid=1015 (administration) groups=1015 (administration)
```



The **multi-user design** ensures security, organization, and controlled access to system resources.



❖ UNIX / LINUX USERS

❑ Multi-User Environment

◆ Multi-User System :

Several users can work on the same machine simultaneously, each with independent access and privileges.

◆ User Permissions :

Each user has specific permissions that determine :

- Which **programs** they can execute
- Which **files or data** they can read
- Where they can **create or modify** files and directories

◆ Personal Workspace :

Every user has a **home directory** — a private area on the disk with full control:

- Create subdirectories and files
- Install personal applications
- Manage their own configurations

◆ Access Control :

Users can **grant or restrict** access to their files for other users, maintaining privacy and security.



❖ UNIX / LINUX USERS

□ User Groups

◆ Purpose of Groups :

- User groups allow **controlled sharing** of files and resources among specific users.
- Each user can belong to **one or more groups**.

◆ Group Identification :

- Every group is assigned a unique **Group Identifier (GID)**.
- Each file or directory is linked to **exactly one group**.

◆ Access Control :

- File permissions are determined using both the **User ID (UID)** and the **Group ID (GID)**.
- The file owner can decide :
 - ✓ *To grant read / write access to group members.*
 - ✓ *To deny access to users outside the group.*

 **Groups** make it easy to collaborate securely by managing who can access or modify shared resources.

Example — File Listing with **ls -l**

```
admin@hp:~$ ls -l
-rw-r--r-- 1 admin administration 150 Sep 28 22:05 etudiants.csv
```



❖ UNIX / LINUX USERS

❑ Superuser (root)

◆ Definition

A **multi-user operating system** includes a special account called the **superuser** (or **root**) — the user with **complete control** over the system.

◆ Main Privileges :

- ❖ **Full access** to all system resources (no restrictions).
- ❖ **Account management:** create, modify, or delete users.
- ❖ **Software installation and system updates.**
- ❖ **Maintenance operations:** backups, recovery, troubleshooting.
- ❖ **Ownership:** has access to **all files and processes** in the system.

⚠ **Caution :** Use the **superuser** account **carefully** — any incorrect command can impact the **entire operating system**.



❖ UNIX / LINUX USERS

□ User Organization

◆ Multi-user System

Linux is a **multi-user operating system** — multiple users can work **simultaneously** on the same system.

◆ Types of Users

Regular User :

- Has **limited rights**.
- Owns a **personal workspace** (home directory) for files and configurations.
- Can **read and execute** most programs but cannot modify system settings.

Administrator (root) :

- Has **all privileges**.
- Responsible for **user management, system configuration, software installation, and maintenance**.

◆ Command Prompts :

– Command executed **with root privileges** (as **root** or via **sudo**).
\$ – Command executed as a **regular user** (non-privileged)



❖ UNIX / LINUX USERS

□ UNIX/Linux users

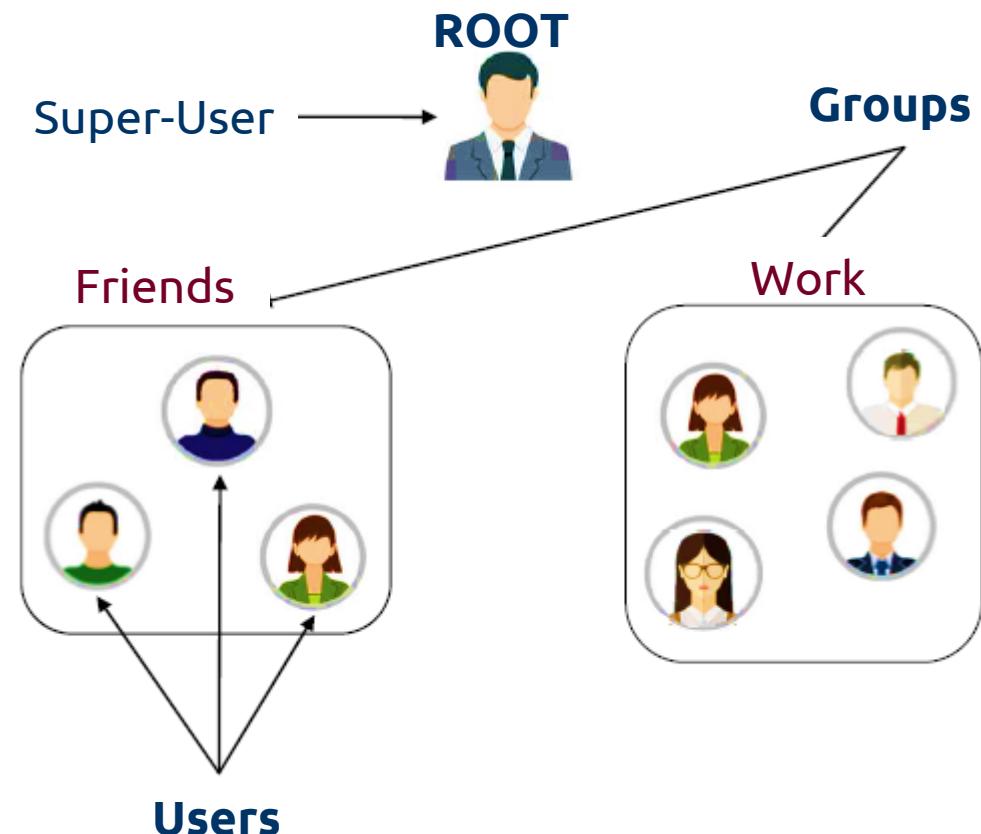
◆ **Organization of Users** : All users are **organized into groups** for better management of permissions and access control.

◆ Group Membership Rules

- Each user must belong to at least one group.
- A user can be a member of **multiple groups** at the same time.
- Every user has a **primary (login) group**, which is assigned by default at login.

◆ Why Groups Matter ?

- Groups help **simplify permission management**.
- They enable **collaboration** by allowing specific users to share files and resources securely.



💡 Grouping users ensures both flexibility and security in a multi-user Linux environment.



❖ UNIX / LINUX USERS

□ Basic Commands

Control System	
<u>poweroff</u>	Shuts down the system
<u>reboot</u>	Restarts the system
<u>shutdown</u>	Shuts down or reboot the system
User Management	
<u>useradd</u>	Create a new user account
<u>userdel</u>	Delete a user account
<u>passwd</u>	Change a user's password
<u>groupadd</u>	Add a new group to the system
<u>groupdel</u>	Delete an existing group from the
id	Display the user and group IDs associated with the current user
<u>su</u>	Switch to another user account

□ Advanced Commands

User Management	
<u>last</u>	Displays the list of previously logged in users
<u>adduser</u>	Creates a new user on the system
<u>usermod</u>	Modifies attributes of an existing user
<u>deluser</u>	Deletes an existing user from the system
<u>delgroup</u>	Deletes an existing group from the system
<u>groups</u>	Prints group membership of a user
...	





المدرسة الوطنية العليا في الأمن السيبراني
NATIONAL SCHOOL OF CYBERSECURITY





THANK YOU for your attention!



Questions ?



المدرسة الوطنية العليا في الأمان السيبراني
NATIONAL SCHOOL OF CYBERSECURITY



For more information about my research works, **Contact Information:**

Dr. Sassi BENTRAD

LISCO Laboratory : <http://lisco.univ-annaba.dz/>

☎ : +213 ...

✉ : sassi.bentrad.enscs@gmail.com // sassi.bentrad@enscs.edu.dz

LinkedIn : www.linkedin.com/in/sassi-bentrad/

Website : <http://www.bentrad-sassi.sitew.com/>

ORCID

Connecting Research
and Researchers

ID : orcid.org/0000-0002-7458-8121

RESEARCHERID : [A-9442-2013](https://publons.com/researcher/A-9442-2013)

SCOPUS Author ID : [44461052600](https://www.scopus.com/authid/detail.uri?authorId=44461052600)

